SpyWolf

Project Audit

Project:

Crootball

May 17, 2022

Overview

This audit has been prepared for **Crootball** to review the main aspects of the project to help investors make an informative decision in the research process.

You will find a a summarized review of the following main key points:

- Contract's source code
- Project and team
- Website
- Social media & online presence

NOTE: We ONLY consider a project safe if they receive our "Certificate of Trust" NFT. This report only points out any potential red flags found in our analysis. Always do your own research before investing in a project.

Smart Contract Review

The contract review process pays special attention to the following:

- Testing the smart contracts against both common and uncommon vulnerabilities
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Ensuring contract logic meets the specifications and intentions of the client.
- Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders.
- Thorough line-by-line manual review of the entire codebase by industry experts.

```
const events = [
    'dragenter',
    'dragleave',
    'dragover', // to allow drop
    'drop'
];
events.forEach(e => {
  fileDropZone.addEventListener(e, (ev) => {
    ev.preventDefault();
    if (ev.type === 'dragenter') {
      fileDropZone.classList.add('solid-border'
    }
    if (ev.type === 'dragleave') {
      fileDropZone.classList.remove('solid-b
    }
    if(ev.type === 'drop') {
      fileDropZone.classList.remove('sol
      handleFiles(ev.dataTransfer.files
        .then(values => values.map(tag
          tag.setAttribute('class',
          fileDropZone.appendChild(
```

"The results of this audit are purely based on the team's evaluation and does not guarantee nor reflect the projects outcome and goal"
- SpyWolf Team

Smart Contract Summary

Contract Name	CROOTBALL
Ticker	CROL
Contract	0x5C0e449DC9f259C3e411A1d566cC73719F5f2E87
Network	Binance smart chain
Language	Solidity
Tax	**Buy: 8%** **Sell: 8%** **Transfer: 100%**
Total Supply	10,000,000,000
Status	Not launched

Current stats

Burn	No burnt tokens
LP Address	Liquidity not added yet
Liquidity	Liquidity not added yet
MaxTxAmount	100,000,000

Issues Checking Status	
Design Logic	Passed ✓
Compiler warnings.	Passed ✓
Private user data leaks	Passed ✓
Timestamp dependence	Passed ✓
Integer Overflow and Underflow	Passed ✓
Race conditions and Reentrancy. Cross-function race conditions	Passed ✓
Possible delays in data delivery	Passed ✓
Oracle calls	Passed ✓
Front running	Passed ✓
DoS with Revert	Passed ✓
DoS with block gas limit	Passed ✓
Methods execution permissions	Passed ✓
Economy model	Passed ✓
The impact of the exchange rate on the logic	Passed ✓
Malicious Event log	Passed ✓
Scoping and Declarations	Passed ✓
Uninitialized storage pointers	Passed ✓
Arithmetic accuracy	Passed ✓
Cross-function race conditions	Passed ✓
Safe Zeppelin module	Passed ✓
Fallback function security	Passed ✓

Featured Wallets

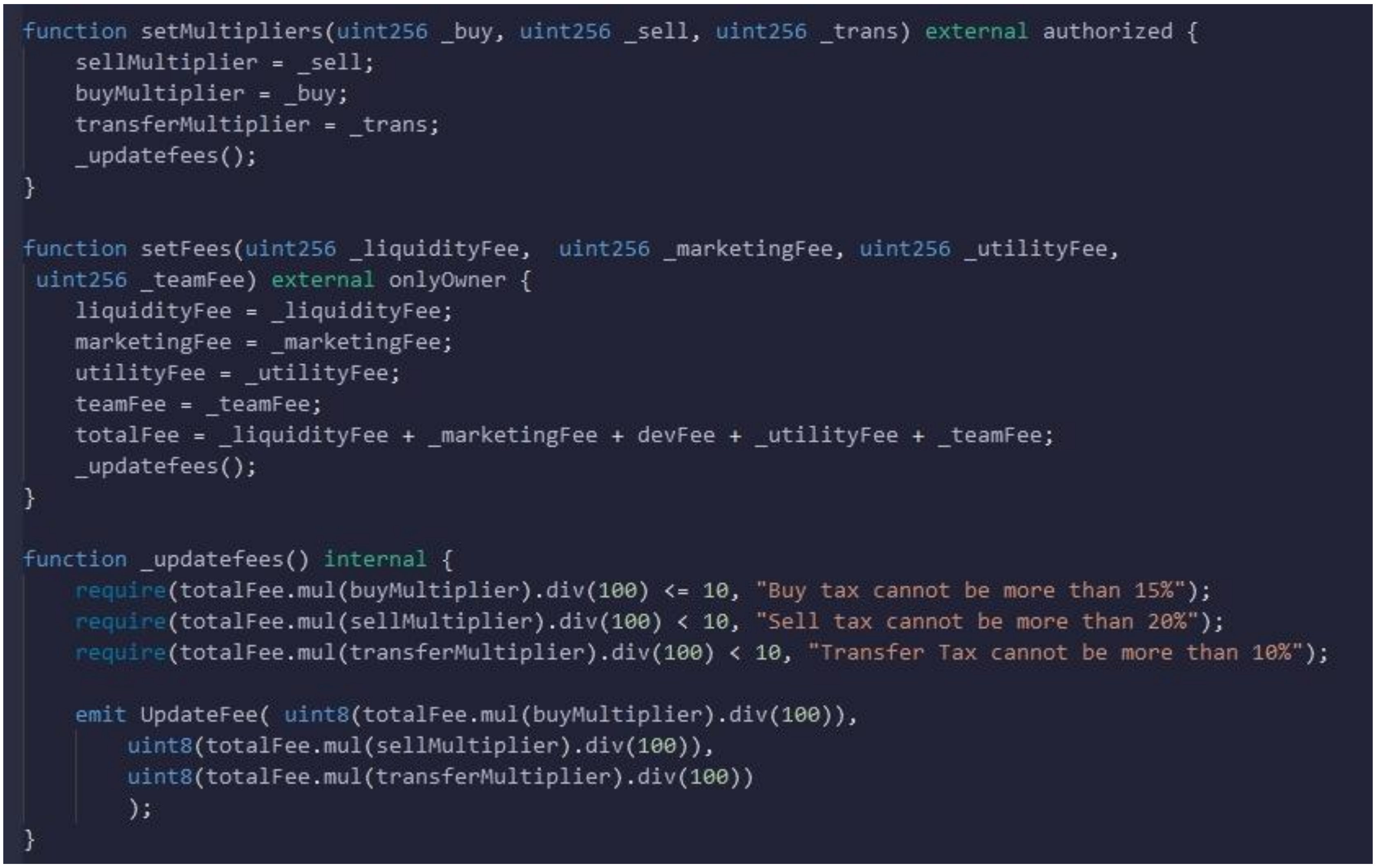
Owner address	0x988afE0379c44b04cb54F40aeDf1089EF04A2985
Utility fee receiver	Same as owner
Team fee receiver	0xD79dc35E7bbdC79CC518D4C9905E30e5Acdf54F7
Marketing fee receiver	0x525B6CDaA9F742A4E7c81AaECa3B3078769C2F6A
LP address	Liquidity not added yet

Top 3 Unlocked Wallets

Wallet 1 (100%)	Same as owner

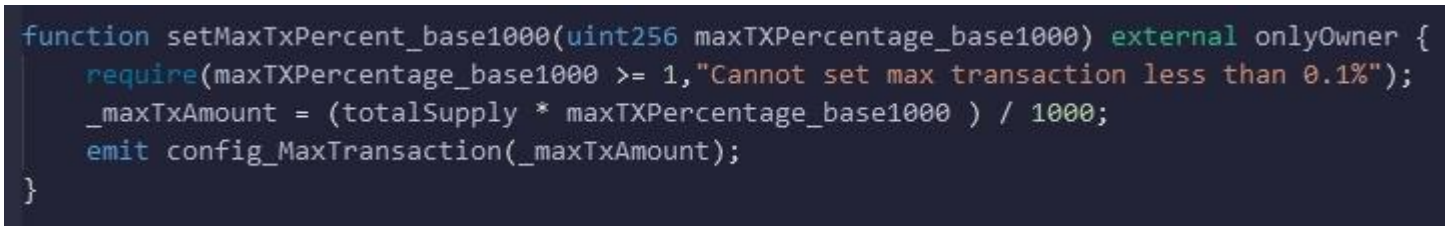
Security Threats

Owner can change buy/sell/transfer fees up to 10%/9%/9%.



```
function setMultipliers(uint256 _buy, uint256 _sell, uint256 _trans) external authorized {
    sellMultiplier = _sell;
    buyMultiplier = _buy;
    transferMultiplier = _trans;
    _updatefees();
}

function setFees(uint256 _liquidityFee,  uint256 _marketingFee, uint256 _utilityFee,
 uint256 _teamFee) external onlyOwner {
    liquidityFee = _liquidityFee;
    marketingFee = _marketingFee;
    utilityFee = _utilityFee;
    teamFee = _teamFee;
    totalFee = _liquidityFee + _marketingFee + devFee + _utilityFee + _teamFee;
    _updatefees();
}

function _updatefees() internal {
    require(totalFee.mul(buyMultiplier).div(100) <= 10, "Buy tax cannot be more than 15%");
    require(totalFee.mul(sellMultiplier).div(100) < 10, "Sell tax cannot be more than 20%");
    require(totalFee.mul(transferMultiplier).div(100) < 10, "Transfer Tax cannot be more than 10%");

    emit UpdateFee( uint8(totalFee.mul(buyMultiplier).div(100)),
        uint8(totalFee.mul(sellMultiplier).div(100)),
        uint8(totalFee.mul(transferMultiplier).div(100))
        );
}
```

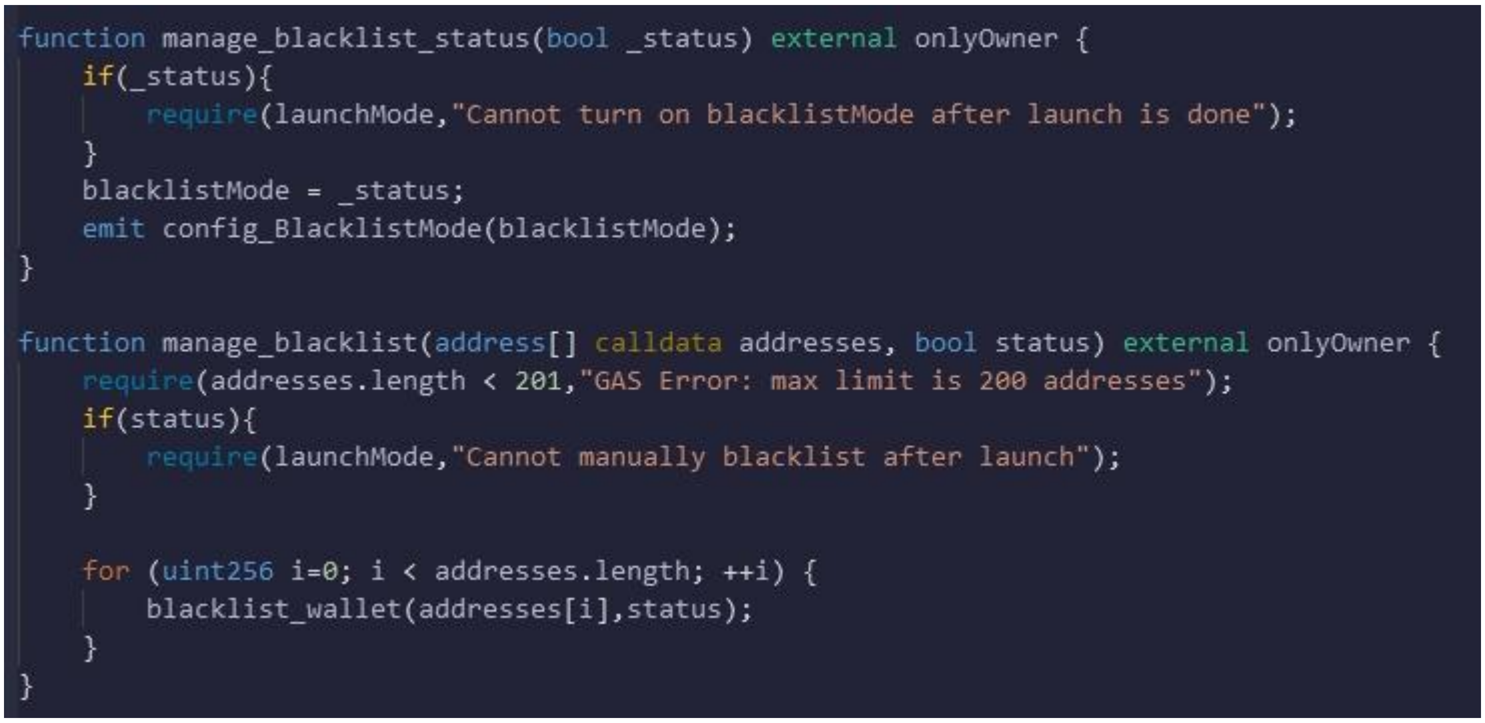
Owner can change max transaction limit, but can’t lower it than 0.1% of total supply.



```
function setMaxTxPercent_base1000(uint256 maxTXPercentage_base1000) external onlyOwner {
    require(maxTXPercentage_base1000 >= 1,"Cannot set max transaction less than 0.1%");
    _maxTxAmount = (totalSupply * maxTXPercentage_base1000 ) / 1000;
    emit config_MaxTransaction(_maxTxAmount);
}
```

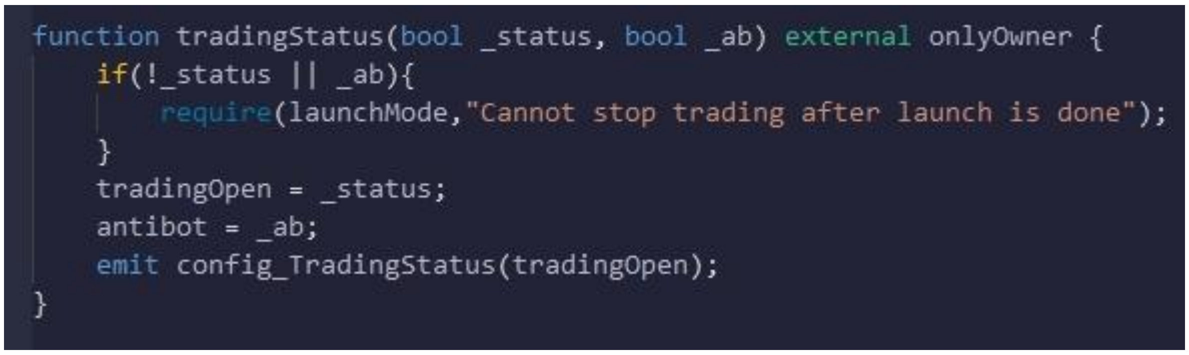
Security Threats

Owner can blacklist address, making it impossible to sell.
This function can be triggered only when ‘launchmode ’ variable is true
Once ‘launchmode ’ becomes false, this function can not be used anymore.



```
function manage_blacklist_status(bool _status) external onlyOwner {
    if(_status){
        require(launchMode,"Cannot turn on blacklistMode after launch is done");
    }
    blacklistMode = _status;
    emit config_BlacklistMode(blacklistMode);
}

function manage_blacklist(address[] calldata addresses, bool status) external onlyOwner {
    require(addresses.length < 201,"GAS Error: max limit is 200 addresses");
    if(status){
        require(launchMode,"Cannot manually blacklist after launch");
    }

    for (uint256 i=0; i < addresses.length; ++i) {
        blacklist_wallet(addresses[i],status);
    }
}
```

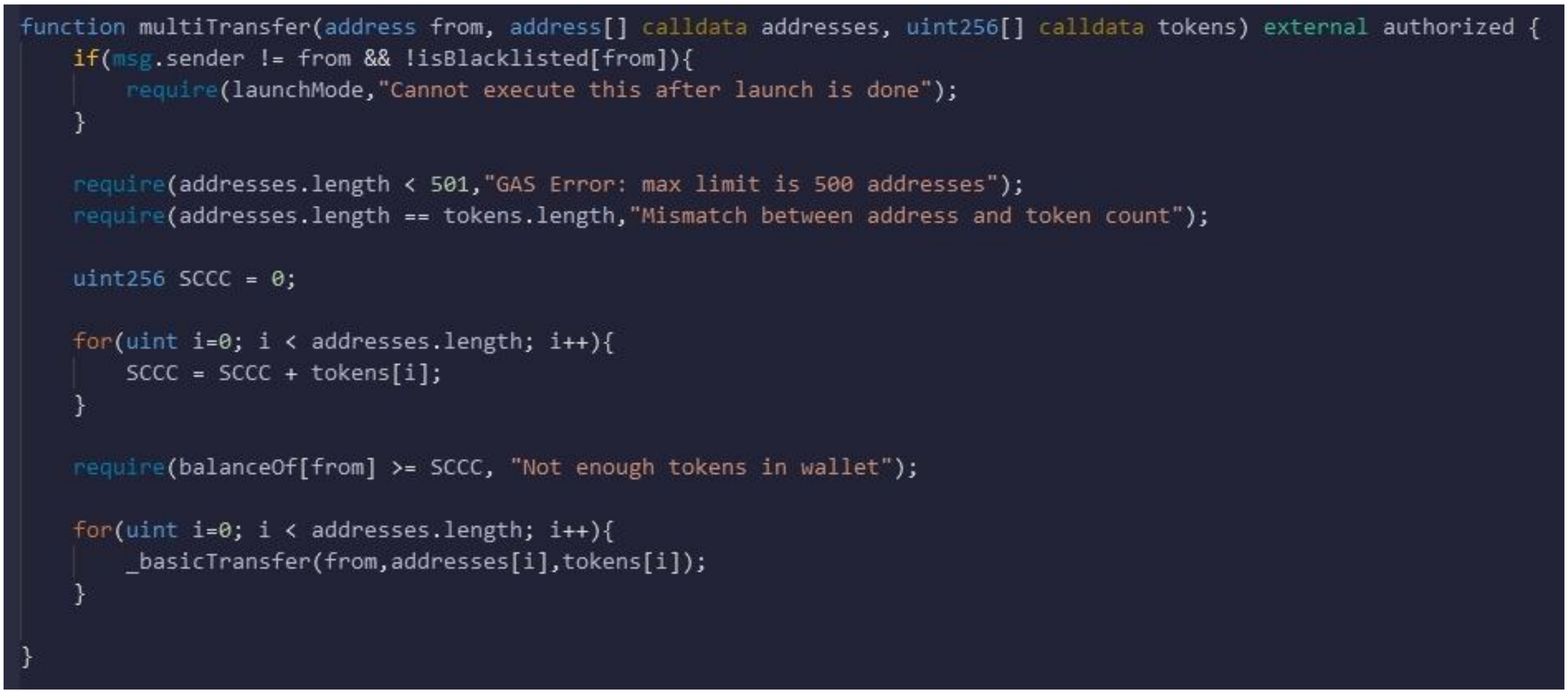
Owner can disable trade, making it impossible to sell.



```
function tradingStatus(bool _status, bool _ab) external onlyOwner {
    if(!_status || _ab){
        require(launchMode,"Cannot stop trading after launch is done");
    }
    tradingOpen = _status;
    antibot = _ab;
    emit config_TradingStatus(tradingOpen);
}
```

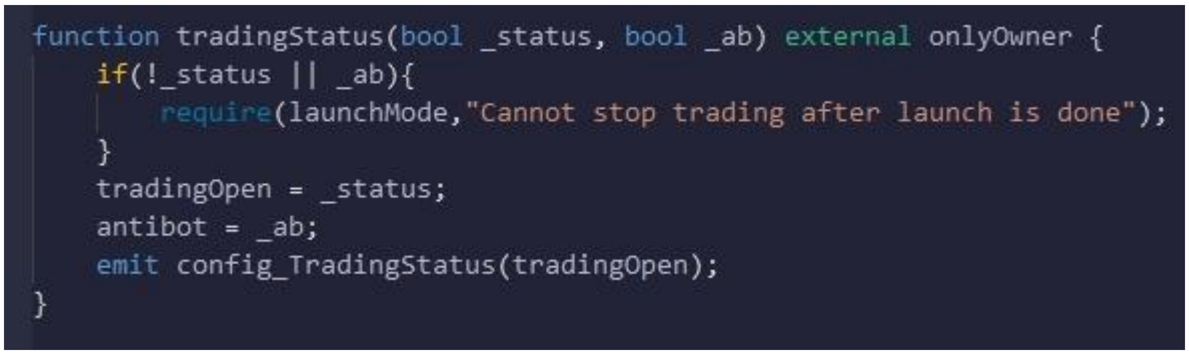
Security Threats

⚠️ **Owner can withdraw tokens from any address, including LP. This function can be triggered only when launchmode is true. Once launchmode becomes false, this function can not be used anymore.**



```
function multiTransfer(address from, address[] calldata addresses, uint256[] calldata tokens) external authorized {
    if(msg.sender != from && !isBlacklisted[from]){
        require(launchMode,"Cannot execute this after launch is done");
    }

    require(addresses.length < 501,"GAS Error: max limit is 500 addresses");
    require(addresses.length == tokens.length,"Mismatch between address and token count");

    uint256 SCCC = 0;

    for(uint i=0; i < addresses.length; i++){
        SCCC = SCCC + tokens[i];
    }

    require(balanceOf[from] >= SCCC, "Not enough tokens in wallet");

    for(uint i=0; i < addresses.length; i++){
        _basicTransfer(from,addresses[i],tokens[i]);
    }

}
```

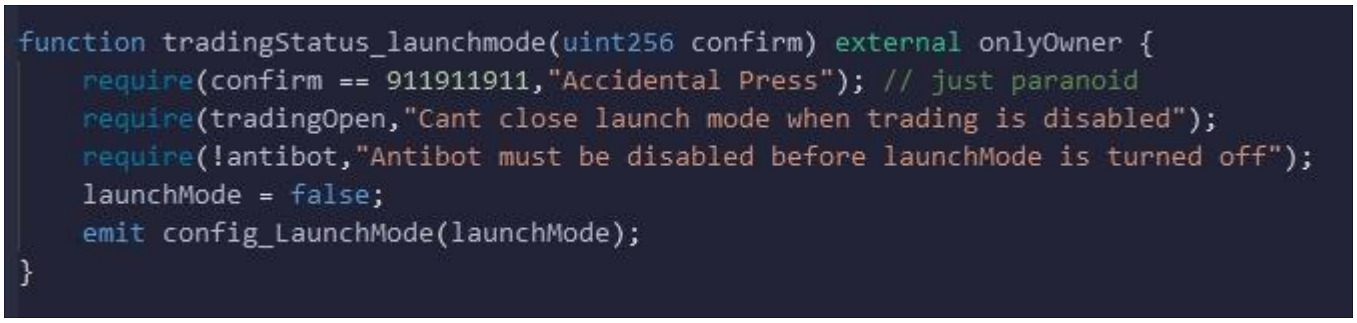
⚠️ **Owner can disable trade, making it impossible to sell.
This function can be triggered only when launchmode is true
Once launchmode becomes false, this function can not be used anymore.**



```
function tradingStatus(bool _status, bool _ab) external onlyOwner {
    if(!_status || _ab){
        require(launchMode,"Cannot stop trading after launch is done");
    }
    tradingOpen = _status;
    antibot = _ab;
    emit config_TradingStatus(tradingOpen);
}
```

Security Threats

Owner can set ‘launchmode’ to false, restricting the above mentioned functions. Current state of the ‘launchmode’ variable is true.
⚠️ This variable is not related to launching of the project itself.



```
function tradingStatus_launchmode(uint256 confirm) external onlyOwner {
    require(confirm == 911911911,"Accidental Press"); // just paranoid
    require(tradingOpen,"Cant close launch mode when trading is disabled");
    require(!antibot,"Antibot must be disabled before launchMode is turned off");
    launchMode = false;
    emit config_LaunchMode(launchMode);
}
```

Tokenomics

⚠️ **There is no information about tokens distribution.**

FootBall Move
Project & Team Review

According to their whitepaper:
Crootball will be online sports betting platform. The project’s team will develop platform for peer to peer betting, betting pools, customizable player and more.

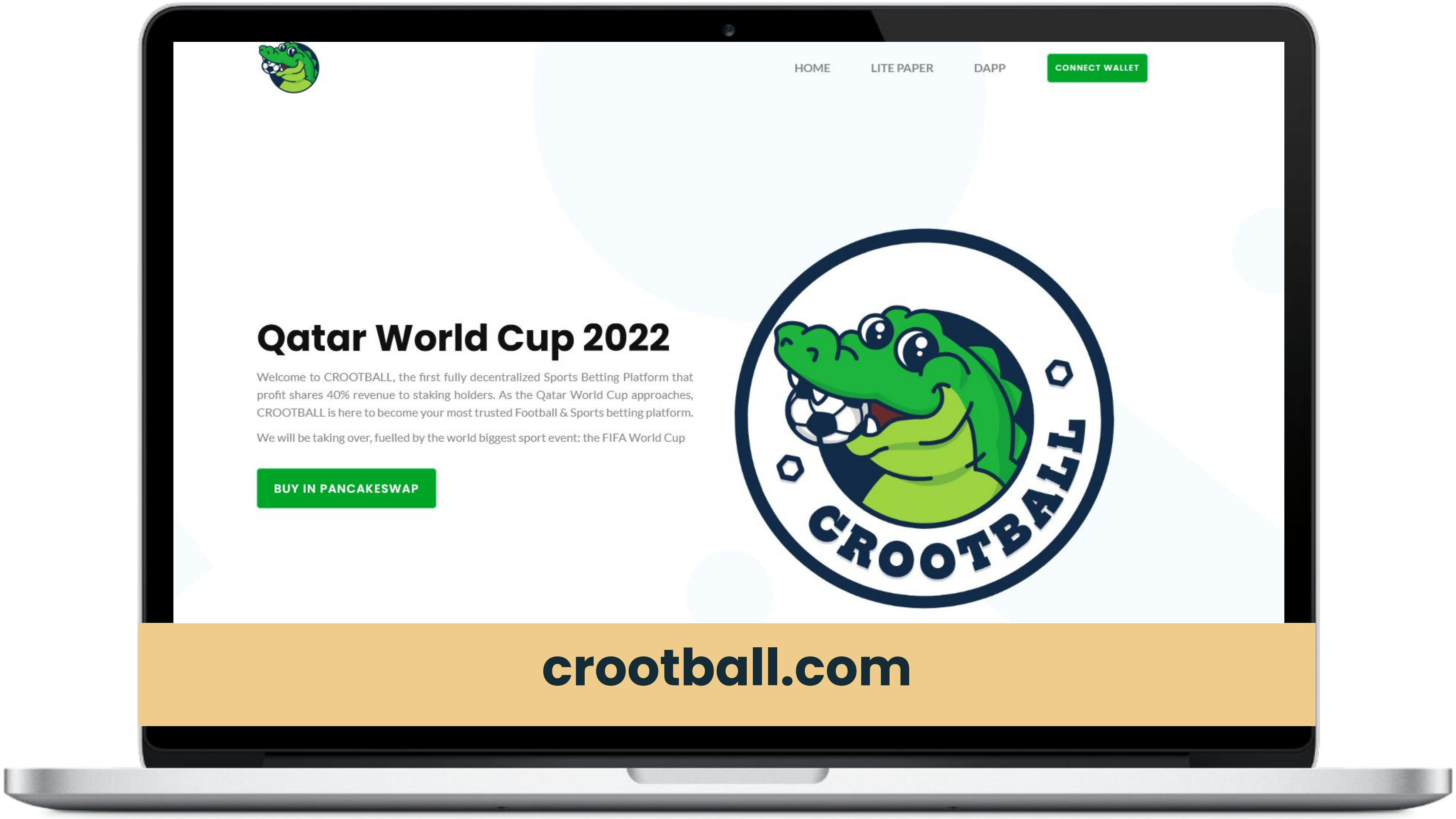
Team:

⚠️ **Team has not been KYC’d** ⚠️

Website Analysis

URL: https://crootball.com/

- **Design:** Pleasant design, single page, appropriate color scheme.
- **Content:** Informative, no grammar mistakes.
- **Whitepaper:** Well written, explanatory.
- **Roadmap:** Goals set at 4 phases with time frames.
- **Mobile-friendly?** Yes
- **Technical:** SSL certificate present. General SEO check passed.



Qatar World Cup 2022

Welcome to CROOTBALL, the first fully decentralized Sports Betting Platform that profit shares 40% revenue to staking holders. As the Qatar World Cup approaches, CROOTBALL is here to become your most trusted Football & Sports betting platform.

We will be taking over, fuelled by the world biggest sport event: the FIFA World Cup

crootball.com

Social Media & Online Presence

Telegram

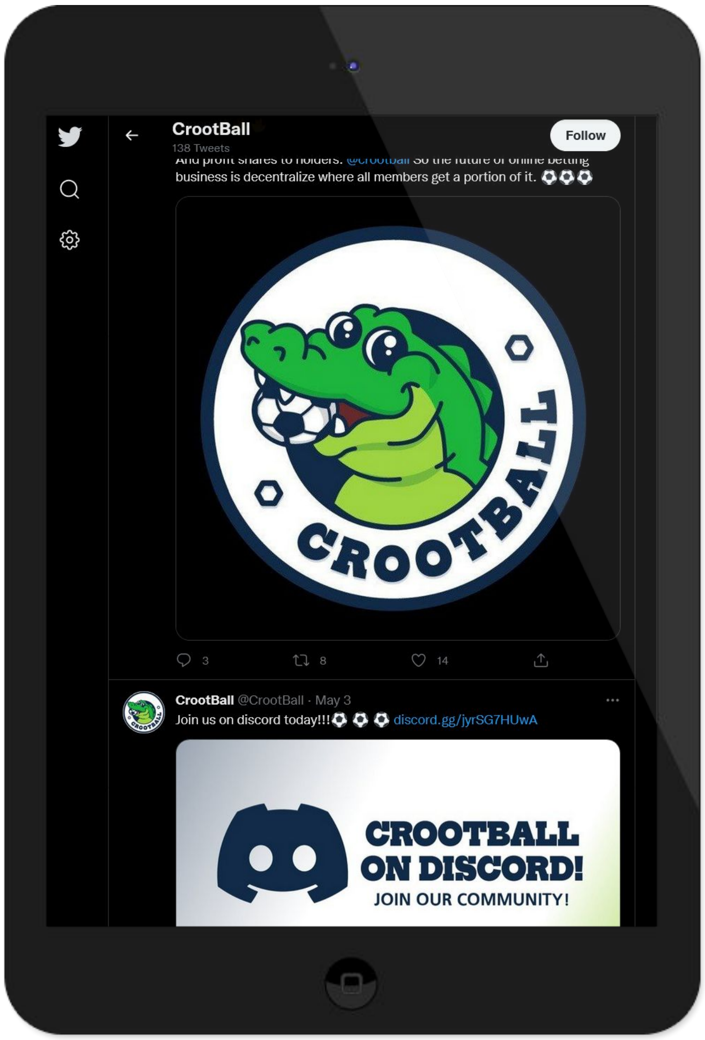
https://t.me/FootballMove

- 426 members
- Active members
- Active mods

Twitter

https://twitter.com/CrootBall

- 354 Followers
- Active



And profit shares to holders. @crootball So the future of online betting business is decentralize where all members get a portion of it. ⚽⚽⚽

CrootBall @CrootBall · May 3

Join us on discord today!!!⚽ ⚽ ⚽ discord.gg/jyrSG7HUwA

About SpyWolf

SpyWolf is a team of crypto security experts that have been performing full audits for projects for the past months in order to ensure safety on the crypto space. Our goal is to help eliminate monetary fraud through our auditing services and utility token, $SPY.

- Website: SpyWolf.co
- Portal: SpyWolf.network
- Telegram: @SpyWolfNework
- Twitter: Twitter.com/SpyWolfNetwork

(Sample Certificate NFT for those who pass audit)

If you are interested in finding out more about our audits and Certificate of Trust NFTs, reach out to contact@spywolf.co.

Disclaimer

This report shows findings based on our limited project analysis, following good industry practice from the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, overall social media and website presence and team transparency details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report.
While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the disclaimer below – please make sure to read it in full.

DISCLAIMER:
By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice.
No one shall have any right to rely on the report or its contents, and SpyWolf and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (SpyWolf) owe no duty of care towards you or any other person, nor does SpyWolf make any warranty or representation to any person on the accuracy or completeness of the report.
The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and SpyWolf hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, SpyWolf hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against SpyWolf, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report. The analysis of the security is purely based on the smart contracts, website, social media and team.
No applications were reviewed for security. No product code has been reviewed.